UNIVERSITAT POLITÈCNICA
DE CATALUNYA

Frictionless grasp with 7 fingers on discretized 3D objects

Máximo Roa, Raúl Suárez

Robòtica Industrial i de Sistemes

IOC-DT-P-2006-19
Juny 2006

Institut d'Organització i Control
de Sistemes Industrials

Frictionless grasps with 7 fingers on Discretized 3D Objects

Máximo Roa*, Raúl Suárez†
Institut of Industrial and Control Engineering (IOC)
Technical Univerity of Catalonia (UPC), Barcelona, Spain
E-mail: maximo.roa@upc.edu, raul.suarez@upc.edu

June 2006

Abstract

This paper presents an algorithm to plan locally optimum frictionless grasps on 3D objects. The objects can be of any arbitrary shape, since the surface is discretized in a cloud of points. The planning algorithm finds an initial force-closure grasp that is iteratively improved through an oriented search procedure. The grasp quality is measured with the "largest ball" criterion, and a force-closure test based on geometric considerations is used. The efficiency of the algorithm is illustrated through numerical examples.

Keywords: Grasp planning, frictionless grasps.

1 Introduction

Grasps capable of ensuring the immobility of the object in front of external disturbances satisfy one of the following properties: form-closure, when the position of the fingers ensures the object immobility, or force-closure, when the forces applied by the fingers ensure the object immobility [1]. Based on any of these properties, the grasp planners calculate the position of the fingers on the object surface. The property to be used depends largely on the field of application: form-closure is used when the task requires a robust grasp not relying on friction, e.g. the fixture of objects to be manufactured or inspected, while force-closure is specially used in grasping and manipulation of objects with a low number of frictional contacts using for instance mechanical grippers or hands. Several algorithms have been developed to determine precision grasps (grasps formed by a set of contact points on the object´s surface) with different number of fingers and satisfying the form or force closure condition in 2D polygonal [2] or non-polygonal [3] objects, 3D polyhedral objects [4, 5] or objects with smooth curved surfaces [6, 7]. However, the development of algorithms to efficiently synthesize grasps in 3D complex real-world objects is still an open research problem.

A widely used technique to represent an arbitrary object is the approximation of the external surface with a triangular mesh of hundreds of faces [8]. The application to these kind of meshes of current algorithms developed for grasp synthesis of polyhedral objects would have a large computational cost. It has been stated that a randomized grasp planner can be quick and efficient to generate good grasps on these objects [9]; the complexity of this grasp planner depends on the object form, not on its number of faces, but the generated grasps are not optimal.

Another approach to deal with complex objects samples the actual surface of the 3D object to generate a set of surface points with their corresponding normal direction. This

*Department of Mechanical and Mechatronic Engineering - National University of Colombia; at present with the IOC-UPC, supported by the Programme Alβan (European Union Programme of High Level Scholarships for Latin America), Scholarship No. E04D039103CO.

†This work was partially supported by the CICYT projects DPI2004-03104 and DPI2005-00112

approach allows the application of search algorithms to objects of arbitrary shape, provided that the number of points is large enough to accurately represent the surface. Wang [10] proposes an algorithm for fixture synthesis; the algorithm finds a fixture minimizing the workpiece positioning errors due to uncertainties in the position of the locators and in the geometry of the workpiece . Ding et al. [11] propose an algorithm to generate a form-closure grasp with seven frictionless contact points; however, it can be trapped in local minima. Liu et al. [12] extend the previous algorithm to find one force-closure (FC) grasp with frictional or frictionless contact points; the algorithm is complete, in the sense that it finds a FC grasp if it exists in the discrete sampling of the surface, but it does not assure any optimality. On the other hand, Niparnan and Sudsang [13] generate a number of 4-finger concurrent FC grasps to provide the user with a large set of grasps, so the user can choose an optimum one according to a quality measure appropriate for the particular task.

This paper deals with the problem of finding a locally optimum FC grasp with frictionless contact points (in this case, the FC grasp is also a form-closure grasp). The proposed approach has two main parts; the first part finds an initial FC grasp, and the second one optimizes that grasp. The initial FC grasp is obtained with an algorithm similar to the one proposed in [12], but using a different FC test that decreases the search complexity. The algorithm used to find the locally optimum grasp in the discrete set of points is the main contribution of the paper; the optimization is carried out using the criterion of the "largest ball" [14], unlike [10] that uses the minimization of workpiece positioning errors as the optimality criterion. The "largest ball" criterion is one of the most popular grasp quality measures, as it accounts for the largest perturbation wrench that the grasp can resist, with independence of its direction.

After this Introduction the article is structured as follows. Section 2 defines the problem to be solved and outlines the two phases of the approach. Section 3 presents the algorithm used to obtain an initial force-closure grasp (first phase), and Section 4 presents the algorithm to optimize the initial grasp (second phase). The algorithms have been implemented and Section 5 shows two results of their application to two objects. Finally, Section 6 presents the conclusions and summarizes future works.

2 Problem overview

2.1 Problem definition

The problem to be tackled is the search of a FC frictionless grasp, locally optimum according to the criterion of the "largest ball", in a set of points representing the external surface of an arbitrary 3D object. The work relies on the following assumptions:

- The contacts between the fingers and the object are frictionless point contacts.
- The external surface of the object is represented with a large set Ω of points, described by position vectors $\boldsymbol{p}_i$ measured with respect to a reference system located in the center of mass (CM) of the object. Each point has an associated normal direction $\hat{\boldsymbol{n}}_i$ aiming to the interior of the object.
- The number of points in Ω is large enough to accurately represent the surface of the object.

2.2 Strategy of solution

The approach proposed in this paper to find a locally optimal FC grasp of a discretized 3D object consists of two phases, including:

1. An algorithm to search for an initial FC grasp from the set of points Ω.
2. An optimization algorithm that begins with the FC grasp obtained in the previous phase, and optimizes it according to the "largest ball" criterion.

Section 3 and Section 4 explain the two phases in detail.

2.3 Frictionless grasps

Seven frictionless contacts are necessary and sufficient to hold a 3D object with a FC grasp, provided that the object has no rotational symmetries [15, 16]. However, note that the rotational symmetry is not an actual limitation for discrete objects, because they are equivalent to a polyhedron with a large number of faces.

With frictionless contact points, the grasp forces can only be applied in the direction normal to the object surface. A force $\boldsymbol{f}_i = \alpha_i \hat{\boldsymbol{n}}_i$ applied on the object at the point $\boldsymbol{p}_i$ generates a torque $\boldsymbol{\tau}_i = \boldsymbol{p}_i \times \boldsymbol{f}_i$ with respect to CM; α_i is a nonnegative value representing the magnitude of the grasping force. The force and the torque are grouped together in a wrench vector (also known as generalized force vector) given by

$$\tilde{\boldsymbol{\omega}}_i = \begin{pmatrix} \boldsymbol{f}_i \\ \boldsymbol{\tau}_i \end{pmatrix} = \alpha_i \begin{pmatrix} \hat{\boldsymbol{n}}_i \\ \boldsymbol{p}_i \times \hat{\boldsymbol{n}}_i \end{pmatrix} \tag{1}$$

The wrenches applied through the contact points on the object can be grouped in a wrench matrix $W = (\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \ldots, \boldsymbol{\omega}_7) \in \Re^{6\times 7}$, where each $\boldsymbol{\omega}_i$, $i = 1, \ldots, 7$, is called a primitive contact wrench when $\alpha_i = 1$ in equation (1). Each physical point $\boldsymbol{p}_i$ in the set Ω has a corresponding wrench $\boldsymbol{\omega}_i$ in the generalized force space; both of them will be called as a grasp point.

3 First phase: getting one force-closure grasp

3.1 Outline of the algorithm

The main ideas of the algorithm used in this Section have a close similarity to those used in [12]. However, the FC test is different, and the search procedure has some variations to overcome the local minima problem present in [12].

The algorithm generates an initial grasp G^1 selecting seven random points from Ω; conforms the corresponding wrench matrix W^1 and checks whether the points form a FC grasp. If they do, then the algorithm finishes. If G^1 is not a FC grasp, then an oriented search is executed, based on separating hyperplanes that define a subset Ω_C^1 containing candidate points to replace one of the current points in G^1. The main steps in the search algorithm are:

1. Generate a random initial grasp $G^k = \{\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_7\}$, $k = 1$.
2. Form the corresponding wrench matrix W^k.
3. Check whether G^k is a FC grasp; if so, the algorithm finishes and returns G^k.
4. Create the subset G_R^k of grasp points in G^k that may be replaced.
5. Create the subset Ω_C^k with candidate points.
6. Find the best possible replacement. Update the counter, $k = k + 1$, and compute the new G^k. Go back to step 2.

The FC test and the search and replacement procedure are explained in detail in the following Subsections.

3.2 Force-closure test

Several criteria have been proposed to test the force-closure property in a particular grasp. A necessary and sufficient condition for the existence of a FC grasp is that the origin of the wrench space lies strictly inside the convex hull (CH) of the primitive contact wrenches [15, 17]. Another qualitative test is given in [18], based on linear matrix inequalities that deal efficiently with frictional constraints, thus avoiding the linearization of the friction cones. The FC test used in [12] stands that querying whether the origin lies inside the CH is equivalent to a ray-shooting problem, solved as a linear programming problem [19]. The FC test used in this work is based on the following lemma.

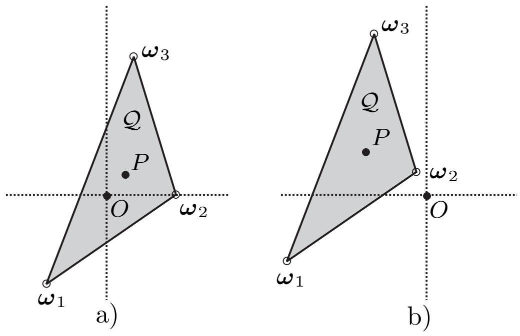


Figure 1: Illustration of the force-closure test in a two-dimensional wrench space: a) Force-closure grasp; b) Non force-closure grasp (P and O lie in different sides of $\overline{\boldsymbol{\omega}_1\boldsymbol{\omega}_2}$).

Lemma 1: Let G be a grasp with a matrix W containing the primitive contact wrenches, and let $\mathcal{Q}$ be the set of strictly interior points of $CH(W)$. The grasp G is FC iff all the boundary hyperplanes of $CH(W)$ leave the set $\mathcal{Q}$ and the origin O of the wrench space in the same half-space.

Proof. Any hyperplane H in the wrench space divides it in two half-spaces, denoted by H^+ and H^-. By definition, the set $\mathcal{Q}$ is fully contained in one of the half-spaces defined by a boundary hyperplane of $CH(W)$. If at least one of the boundary hyperplanes has the origin on one half-space and the set $\mathcal{Q}$ in the other, then the origin O lies outside $CH(W)$ and the grasp is not FC. If the origin O lies on the same side than $\mathcal{Q}$ for all the boundary hyperplanes, then $O \in \mathcal{Q}$, i.e. the origin lies in the convex hull $CH(W)$, and the grasp is FC. $\square$

Note that only one point $P \in \mathcal{Q}$ is enough to prove whether O lies inside $CH(W)$, because the whole set $\mathcal{Q}$ is fully contained on one side of any boundary hyperplane. A strictly positive combination of the primitive contact wrenches must be an interior point of $CH(W)$; therefore, P is chosen as the centroid of the primitive contact wrenches:

$$P = \frac{1}{7}\sum_{i=1}^{7} \boldsymbol{\omega}_i \tag{2}$$

Then, the FC test verifies if the centroid P and the origin O lie on the same side for all the boundary hyperplanes of $CH(W)$. Fig. 1 illustrates Lemma 1 with an example in a hypothetical two-dimensional wrench space (the actual wrench space is 6-dimensional).

The ray-shooting test used in [12] requires to solve a linear programming problem in each call of the FC test; the test based on Lemma 1 is computationally less complex, as it just finds the boundary hyperplanes and checks whether the centroid P and the origin O lie on the same side of these planes simply checking a linear equation. The computational costs due to the calculation of $CH(W)$ and P are equal in both algorithms.

3.3 Search procedure

If the grasp G^k fails the FC test, the search procedure iteratively tries to improve the grasp by changing one of the points in G^k, looking for a reduction in the distance between $CH(W)$ and the origin O; that is the same key idea used in [12], although some variations are introduced here to overcome the local minimum problem. The procedure consists of three steps.

1. The first step is the determination of the subset G^k_R of grasp points in G^k that may be replaced. If there are several boundary hyperplanes that produce the FC test failure, hereafter called "critical hyperplanes", then the points to be replaced are the common points to all the critical hyperplanes. If there is just one critical hyperplane, then G^k_R

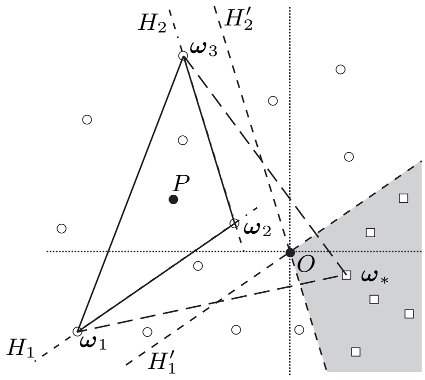


Figure 2: Non force-closure grasp. Wrenches in the gray zone belong to Ω_C^k. The CH using a candidate point $\boldsymbol{\omega}_*$ is also shown.

includes all of the points defining the facet contained in that hyperplane. Fig. 2 shows a non FC grasp; two hyperplanes, H_1 and H_2, produce the FC test failure. The set of possible points to be replaced is $G_R^k = \{\boldsymbol{\omega}_2\}$.

2. The second step is the generation of the subset Ω_C^k with the candidate points to replace one of the points in G_R^k. This subset is determined using hyperplanes parallel to the critical hyperplanes and passing through the origin; the candidate points will be those than simultaneously lie in the opposite side of P with respect to those hyperplanes. In Fig. 2, the hyperplanes H_1' and H_2' define the subset Ω_C^k, represented as a gray zone; wrenches that lie in that zone (depicted as empty squares) belong to Ω_C^k.

3. The third step is the replacement of one of the points in G_R^k with a point from Ω_C^k. A point $\boldsymbol{\omega}_*$ is randomly picked up from Ω_C^k; then, $\boldsymbol{\omega}_*$ replaces the closest point in G_R^k, using the euclidean distance as metric. The candidate grasp CG^* is formed with that replacement (in the example in Fig. 2, $CG^* = \{\boldsymbol{\omega}_1, \boldsymbol{\omega}_*, \boldsymbol{\omega}_3\}$), and the centroid P^* and the distance PO^* are computed. If for any point $PO^* < PO^k$, then the best-first motion is performed, and the corresponding point $\boldsymbol{\omega}_*$ is selected as the replacement point. If all the points in G_R^k have been checked out and none of them decreases the distance PO^k, the replacement is done choosing the candidate CG^* that gives the smaller distance PO^*. Finally, the counter k is updated, and the selected point is included in the new grasp G^k.

   To avoid falling in a local minimum, the generated grasps G^k are stored, and if the third step gives an already considered grasp, then the next best non-visited candidate is taken for the replacement. This consideration allows the grasp search procedure to overcome local minima until a FC grasp is found. In this sense, the algorithm is complete in the discrete domain (as the algorithm in [12] it finds a FC grasp if there is one).

4 Second phase: finding a locally optimum grasp

4.1 Outline of the algorithm

The optimization algorithm begins with an initial FC grasp obtained through the procedure described in Section 3 and the optimization is done according to the "largest ball" criterion [14]. The quality measure is the largest perturbation wrench that the grasp can resist with independence of its direction; geometrically, that quality is equivalent to the radius of the largest ball centered at the origin of the wrench space and fully contained in $CH(W)$, or, in other words, it is also equivalent to the distance from the origin of the wrench space to the closest facet of $CH(W)$.

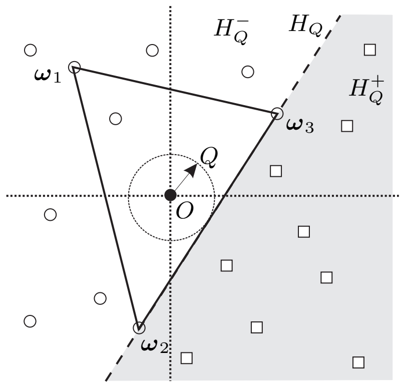


Figure 3: Selection of the subset Ω_C^k with the candidate points that may improve the grasp quality.

Beginning with the initial grasp G^1, the algorithm searches the facet of $CH(W^1)$ closest to the origin, hereafter F_Q, and computes its distance to the origin, i.e. the quality of the current grasp, Q^1. Then, the algorithm selects the candidate points to replace one of the points in F_Q using a separating hyperplane, in a similar way as in the algorithm presented in Section 3. Each one of the candidate points generates 6 candidate grasps, $CG_i^*, i = 1, \dots, 6$, by replacing each one of the vertex points defining the facet F_Q. Picking one candidate point at a time, the corresponding 6 candidate grasps are checked for the FC property. For each FC candidate grasp, the grasp quality is computed; if that quality is greater than Q^1, the corresponding candidate grasp is selected and G, W and Q are updated. The procedure continues until no further improvement can be achieved, i.e. until the algorithm finds a local minimum.

The main steps in the algorithm are:

1. Find an initial FC grasp, $G^k = \{\boldsymbol{\omega}_1, \dots, \boldsymbol{\omega}_7\}$, $k = 1$, using the algorithm presented in Section 3.

2. Determine F_Q, the facet of the convex hull $CH(W^k)$ closest to the origin, and compute the grasp quality Q^k.

3. Create the subset Ω_C^k with the candidate points that may produce an improvement in the grasp.

4. Picking one point ω_* from Ω_C^k, create the candidate grasps $CG_i^*, i = 1, \dots, 6$. For the candidate grasps that are FC grasps compute the expected grasp quality Q^*. When $Q^* > Q^k$, select that candidate grasp, update the counter $k = k + 1$, and G^k and W^k, and go back to step 2. If there is no improvement in Q^k once all the points in Ω_C^k have been considered, then a local minimum has already been reached, and the algorithm finishes.

Steps 3 and 4 are fully described in the following Subsection.

4.2 Optimization procedure

The initial grasp in the optimization procedure is a FC grasp, so the origin of the wrench space lies inside $CH(W)$ (as illustrated in Fig. 3 for a hypothetical two-dimensional wrench space). The grasp quality Q is fixed by F_Q, the closest facet of $CH(W)$ to the origin. The subset Ω_C^k with the points that may improve the grasp quality is defined using H_Q, the hyperplane containing the facet F_Q. The origin O will be located in one of the half-spaces defined by H_Q, e.g. H_Q^-, and the candidate points will be all those lying in the opposite open half-space, i.e. H_Q^+ in the example.

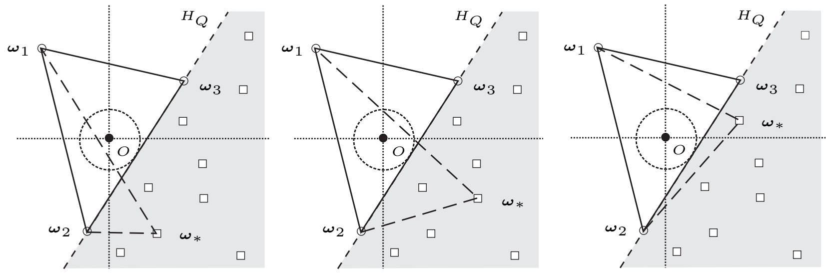


Figure 4: Possible cases in the optimization algorithm: a) Non-feasible candidate grasp, b) Discarded candidate grasp, c) Feasible candidate grasp.

By the selection procedure, all the wrenches $\omega^* \in \Omega_C^k$ are external points to $CH(W)$. When replacing one vertex $\boldsymbol{\omega}_i$ from the actual CH with the candidate wrench $\boldsymbol{\omega}_*$, the latter will be a vertex of the new CH. Using this property, to check if a candidate grasp CG^* keeps the FC property the verification of *Lemma 1* for the hyperplanes containing the facets of the new CH is sufficient; these facets are constructed from the old ones replacing $\boldsymbol{\omega}_i$ with $\boldsymbol{\omega}_*$ (the explicit computation of the new CH is not required).

This FC verification is carried out for 6 candidate grasps for each candidate point because a nearest-neighbor replacement, similar to the replacement used in Section 3, may leave out some possible FC candidate grasps. For the FC candidate grasps, the expected grasp quality Q^* is computed; if for any candidate grasp $Q^* > Q^k$, then the best-first motion is performed, and the candidate grasp becomes the new grasp G^k. Fig. 4 shows three possible cases when considering the candidate points; case (a) is a non-feasible grasp because it loses the FC property, case (b) is discarded because the grasp has a smaller quality than in the previous grasp, and case (c) is a good grasp that actually improves the grasp quality; thus it becomes the grasp for the next iteration cycle.

5 Examples

The proposed optimization strategy has been implemented using Matlab on a PC with a Pentium IV 3.2 GHz CPU. The performance of the algorithm is illustrated using two discretized objects, a parallelepiped and a knight (chess piece) (Fig. 5); the points $\boldsymbol{p}_i$ describing the surface of the objects are obtained by taking the centroids of each triangle in the mesh.

In the first example, the parallelepiped is described with a mesh of 1628 triangles. Although there are more efficient algorithms to generate a FC grasp on polyhedra, this simple figure makes more difficult the search of the first FC grasp, as the initial randomized grasp may place all the fingers on a single face (because there are two large faces, the probability of placing a finger in those faces is greater). Fig. 6 shows a grasp example; the time elapsed to obtain an initial FC grasp is 1.3 seconds in 11 iterations, and the time to get a locally optimum grasp is 15.5 seconds in 19 iterations. Fig. 6a and 6b show the initial and suboptimum FC grasps. Fig. 6c plots $\|PO\|$ against the iteration number; the distance initially increases, but after that peak it decreases. Fig. 6d plots the grasp quality in the optimization phase; the quality always increases monotonically until it finds the suboptimum grasp.

The locally optimum grasp obtained depends on the initial grasp. For the previous example, the initial grasp quality is 0.008, and the suboptimum gasp has a quality of 0.263; the improvement factor, taken as the ratio of qualities for the optimized grasp and the initial FC grasp, is 32.9. To obtain a better insight into the performance of the algorithm, 50 locally optimum grasps were computed using different random initial grasps. The quality distribution of the initial and suboptimum FC grasps is shown in Fig. 7, and the correlation between initial and final grasp qualities is shown in Fig. 8. The average quality gives an idea

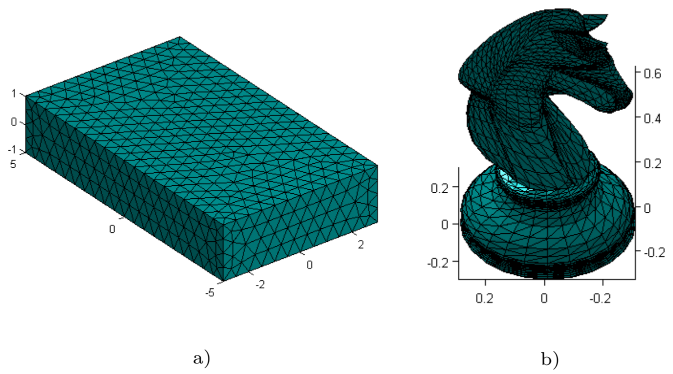


Figure 5: Objects used in the examples: a) Parallelepiped discretized with a mesh of 1628 triangles, b) Knight discretized with 4750 triangles.

of the behavior of the algorithm; it is 0.028 and 0.242 for the initial and locally optimum FC grasps, respectively. The average improvement factor is 8.6, considering an initial grasp given by an algorithm with a random nature such as the presented in Section 3 or in [12].

The knight used in the second example is discretized with 4750 triangles (Fig. 5b). Fig. 9 shows the results for a particular case; the initial grasp is found after 5 iterations in 5.3 seconds, and the suboptimum grasp is obtained after 36 iterations in 70.2 seconds. The grasp qualities are 0.012 and 0.075 for the initial and suboptimum FC grasps, respectively, with an improvement factor of 6.3. Fig. 10 shows the quality distribution for 50 initial and locally optimum grasps, and Fig. 11 shows the correlation between initial and final grasp qualities. The average quality for the initial FC grasp is 0.0025, and 0.069 for the locally optimum grasp. The average improvement factor is 28.2. The results illustrate the relevance and efficiency of the algorithm.

6 Conclusions

This paper proposes a new approach to obtain a locally optimum frictionless grasp in 3D discretized objects. The procedure has two main parts: the first one looks for an initial FC grasp, and its main ideas were presented in [12], although a different and more efficient FC test is used. The second part improves the initial FC grasp through an oriented search procedure, obtaining average improvement factors in the grasp quality above 8, as illustrated by the two examples presented here. The algorithm optimizes the grasp quality according to the "largest ball" criterion, one of the most popular grasp quality measures. Future work includes finding a limit on the maximum grasp quality achievable for a particular object, and developing appropriate algorithms to plan a globally optimum grasp that achieves that quality.

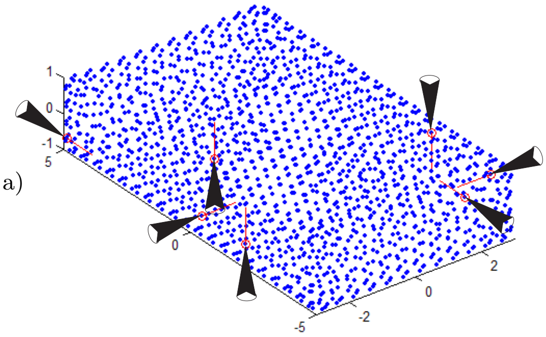
References

[1] A. Bicchi. On the closure properties of robotic grasping. *Int. J. Robotics Research*, 14(4):319–344, 1995.

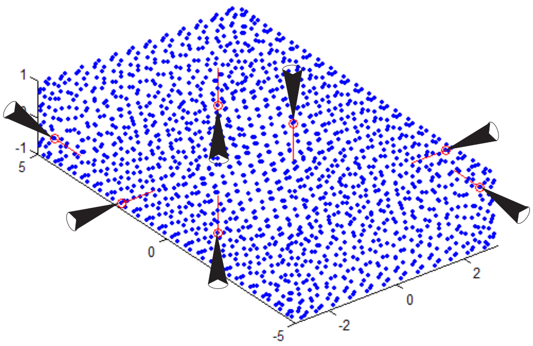
[2] Y.H. Liu. Computing n-finger form-closure grasps on polygonal objects. *Int. J. Robotics Research*, 19(2):149–158, 2000.

[3] J. Cornellà and R. Suárez. On computing form-closure grasps/fixtures for non-polygonal objects. In *Proc. IEEE Int. Symp. Assembly and Task Planning, ISATP 2005*, pages 138–143, 2005.

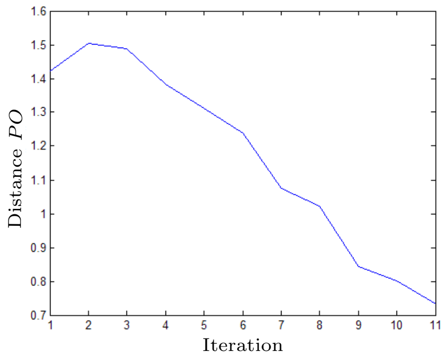
a)



b)



c)



d)

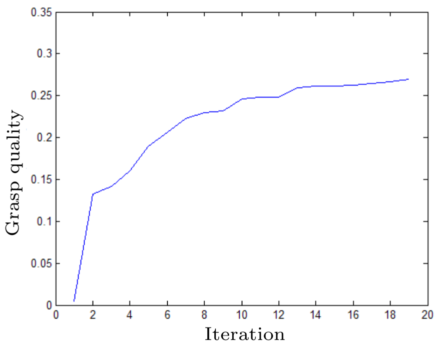


Figure 6: FC grasp on the parallelepiped: a) Initial FC grasp, b) Locally optimum FC grasp, c) Variation in the distance PO, d) Increase in the grasp quality.

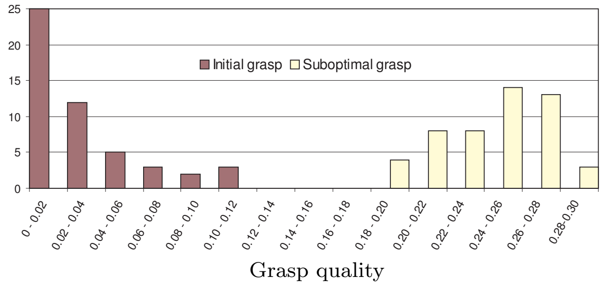


Figure 7: Histograms with the grasp quality distribution for the parallelepiped in the initial and locally optimum FC grasps.

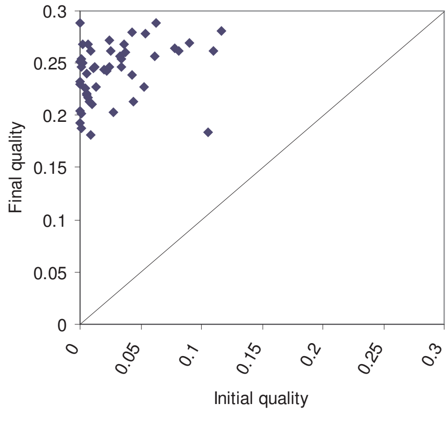
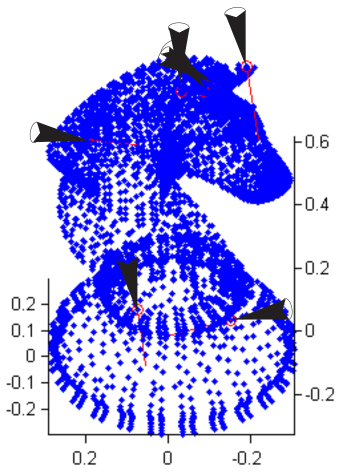
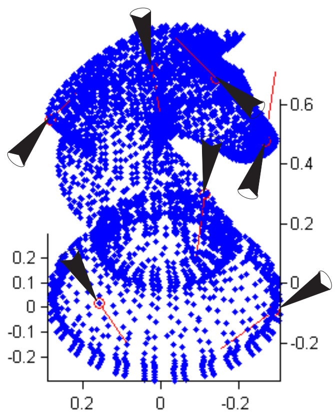


Figure 8: Initial vs. final quality for the parallelepiped grasps.

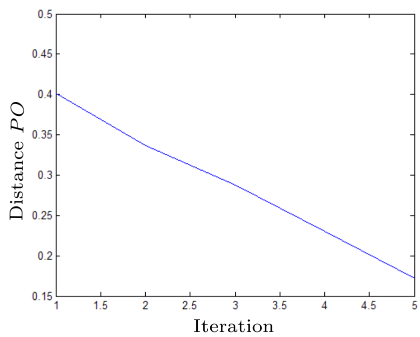
a)



b)



c)



d)

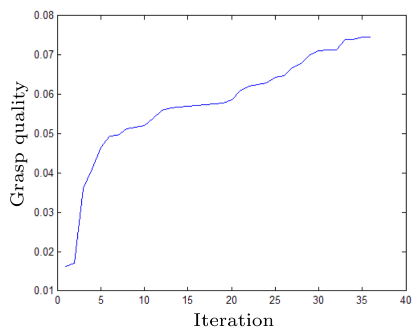


Figure 9: FC grasp on the knight: a) Initial FC grasp, b) Locally optimum FC grasp, c) Decrease in the distance PO, d) Increase in the grasp quality.

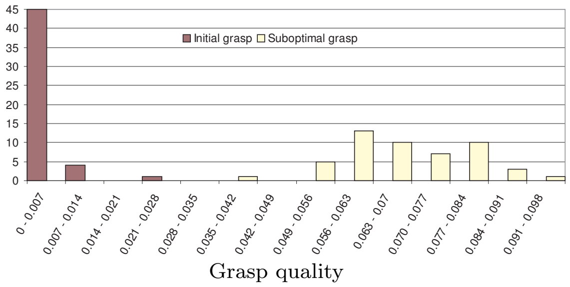


Figure 10: Histograms with the grasp quality distribution for the knight in the initial and locally optimum FC grasps.

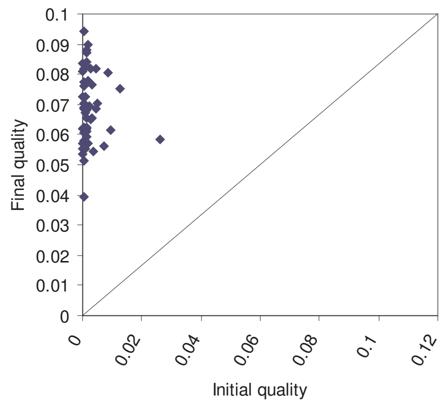


Figure 11: Initial vs. final quality for the knight grasps.

[4] J. Ponce, S. Sullivan, A. Sudsang, J.D. Boissonat, and J.P. Merlet. On computing four-finger equilibrium and force-closure grasps of polyhedral objects. *Int. J. Robotics Research*, 16(1):11–35, 1997.

[5] D. Ding, Y.H. Liu, and S. Wang. Computation of 3-D form-closure grasps. *IEEE Trans. Robotics and Automation*, 17(4):515–522, 2001.

[6] X. Zhu and J. Wang. Synthesis of force-closure grasps on 3-D objects based on the Q distance. *IEEE Trans. Robotics and Automation*, 19(4):669–679, 2003.

[7] X. Zhu and H. Ding. Planning force-closure grasps on 3-D objects. In *Proc. IEEE ICRA 2004*, pages 1258–1263, 2004.

[8] R.J. Campbell and P.J. Flynn. A survey of free-form object representation and recognition techniques. *Computer Vision and Image Understanding*, 81:166–210, 2001.

[9] Ch. Borst, M. Fischer, and G. Hirzinger. Grasping the dice by dicing the grasp. In *Proc. IEEE/RSJ IROS 2003*, pages 3692–3697, 2003.

[10] M.Y. Wang. An optimum design for 3-D fixture synthesis in a point set domain. *IEEE Trans. Robotics and Automation*, 16(6):839–846, 2000.

[11] D. Ding, Y.H. Liu, and M.Y. Wang. On computing inmobilizing grasps of 3-D curved objects. In *Proc. IEEE Int. Symp. on Computational Intelligence in Robotics and Automation*, pages 11–16, 2001.

[12] Y.H. Liu, M.L. Lam, and D. Ding. A complete and efficient algorithm for searching 3-D form closure grasps in the discrete domain. *IEEE Trans. Robotics*, 20(5):805–816, 2004.

[13] N. Niparnan and A. Sudsang. Fast computation of 4-fingered force-closure grasps from surface points. In *Proc. IEEE IROS 2004*, pages 3692–3697, 2004.

[14] C. Ferrari and J. Canny. Planning optimal grasps. In *Proc. IEEE ICRA 1992*, pages 2290–2295, 1992.

[15] B. Mishra, J.T. Schwartz, and M. Sharir. On the existence and synthesis of multifinger positive grips. *Algorithmica*, 2(4):541–558, 1987.

[16] X. Markenscoff, L. Ni, and C.H. Papadimitriou. The geometry of grasping. *Int. J. Robotics Research*, 9(1):61–74, 1990.

[17] R.M. Murray, Z. Li, and S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Ratón, Florida, 1994.

[18] L. Han, J.C. Trinkle, and Z.X. Li. Grasp analysis as linear matrix inequality problems. *IEEE Trans. Robotics and Automation*, 16(6):663–674, 2000.

[19] Y.H. Liu. Qualitative test and force optimization of 3-D frictional form-closure grasps using linear programming. *IEEE Trans. Robotics and Automation*, 15(1):163–173, 1999.